11IDM-0087 / 2011-01-0725

A Simulation and Optimization Methodology for Reliability of Vehicle Fleets

Zissimos P. Mourelatos, Jing Li, Vijitashwa Pandey
Oakland University, Rochester Hills, MI 48309

Amandeep Singh, Matthew Castanier, David Lamb
US Army, TARDEC, Warren, MI 48397

ABSTRACT

Understanding reliability is critical in design, maintenance and durability analysis of engineering systems. A reliability simulation methodology is presented in this paper for vehicle fleets using limited data. The method can be used to estimate the reliability of non-repairable as well as repairable systems. It can optimally allocate, based on a target system reliability, individual component reliabilities using a multi-objective optimization algorithm. The algorithm establishes a Pareto front that can be used for optimal tradeoff between reliability and the associated cost. The method uses Monte Carlo simulation to estimate the system failure rate and reliability as a function of time. The probability density functions (PDF) of the time between failures for all components of the system are estimated using either limited data or a user-supplied MTBF (mean time between failures) and its coefficient of variation. The reliability-cost tradeoff analysis utilizes a user-supplied relationship between reliability and cost for each component. The tradeoff can be used to optimally determine the reliability of each component in order to maximize the system reliability and simultaneously, minimize the acquisition and repair cost for the system. A detailed example highlights the methodology and demonstrates its main features.

1. INTRODUCTION

System reliability analysis is critical to design, preventive maintenance, availability and maintainability through time [1]. Better reliability is usually achieved through better design, better materials or improved manufacturing methods and usually results in an increase in cost. The designer (or the end-user) must take this tradeoff into consideration. In addition, system level reliability targets are usually provided and not the reliability targets of the individual

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
30 NOV 2011	**Journal Article**	**30-11-2011 to 30-11-2011**

4. TITLE AND SUBTITLE **A SIMULATION AND OPTIMIZATION METHODOLOGY FOR RELIABILITY OF VEHICLE FLEETS**	5a. CONTRACT NUMBER **W56HZV-08-C-0236**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Zissimos Mourelatos; Jing Li; David Lamb; Amandeep Singh; Matt Castanier**	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Oakland University,Zissimos Mourelatos,Rochester,MI,48309**	8. PERFORMING ORGANIZATION REPORT NUMBER **; #21373**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **U.S. Army TARDEC, 6501 E.11 Mile Rd, Warren, MI, 48397-5000**	10. SPONSOR/MONITOR'S ACRONYM(S) **TARDEC**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S) **#21373**

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Paper submitted to SAE WORLD CONGRESS 2011

14. ABSTRACT
Understanding reliability is critical in design, maintenance and durability analysis of engineering systems. A reliability simulation methodology is presented in this paper for vehicle fleets using limited data. The method can be used to estimate the reliability of non-repairable as well as repairable systems. It can optimally allocate, based on a target system reliability individual component reliabilities using a multi-objective optimization algorithm. The algorithm establishes a Pareto front that can be used for optimal tradeoff between reliability and the associated cost. The method uses Monte Carlo simulation to estimate the system failure rate and reliability as a function of time. The probability density functions (PDF) of the time between failures for all components of the system are estimated using either limited data or a usersupplied MTBF (mean time between failures) and its coefficient of variation. The reliability-cost tradeoff analysis utilizes a user-supplied relationship between reliability and cost for each component. The tradeoff can be used to optimally determine the reliability of each component in order to maximize the system reliability and simultaneously, minimize the acquisition and repair cost for the system. A detailed example highlights the methodology and demonstrates its main features.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**23**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

components. This leads us to the problem of assigning component level reliability targets based on system targets. This problem is referred to as reliability allocation. It is obvious that many combinations of component level reliabilities (designs) can lead to the same system level reliability. How do we select a design among these that best satisfies the end user requirements? End users are usually concerned with multiple attributes which constrain the design space and help us reduce the number of designs that meet the system reliability targets. Multiple attributes however, increase the computational cost of evaluating designs for fitness to find the optimal one. Optimization methods should be carefully chosen or modified to handle this additional burden. Finally, the methodology for component level reliability allocation should incorporate tradeoff preferences of the end user. It should either incorporate the end user's utility function over the attributes *or* present him/her with a manageable number of designs from which the best one can be chosen based on their tradeoff behavior.

Reliability-cost tradeoffs and component reliability allocation are important in system acquisition and logistics as well as in evaluating new technologies to meet performance targets. The particular application we consider in this paper is reliability-cost study for vehicle fleets. Reliability methods in engineering have existed for some time. Analytical methods include the classical first-order and second-order reliability methods (FORM [2], SORM [3, 4]), and multi-point approximation methods [5]. Integrating reliability into a design problem has been studied under the umbrella term of reliability based design optimization (RBDO). The system is optimized subject to probabilistic constraints which ensure a small probability of violation [6, 7]. Optimization in RBDO is performed using either classical gradient based methods or heuristics like Genetic Algorithms [8]. Time dependence of reliability, is also critical to understanding the performance of real life engineering systems that exhibit degradation. Very few studies have investigated the interplay of reliability allocation in engineering design, time dependence of reliability and cost tradeoffs associated with increasing reliability.

In this paper we implement our methodology on a series system. It is equally applicable to reparable as well as non-reparable systems. For demonstration purposes, we assume that a component failure mileage[1] or time is a Beta distributed random variable [9]. This does not limit our discussion because the family of Beta distributions exhibits a lot of flexibility in modeling different types of distributions. For a given useful mileage, the mean value, standard deviation, failure counts and reliability of a component can be analytically obtained. However, the system reliability depends on the layout of the components, and may be generally estimated using Monte-Carlo Simulation (MCS). The unit cost of a component is assumed to exponentially depend on the mean time between failures (MTBF) while the coefficient of variation is assumed constant.

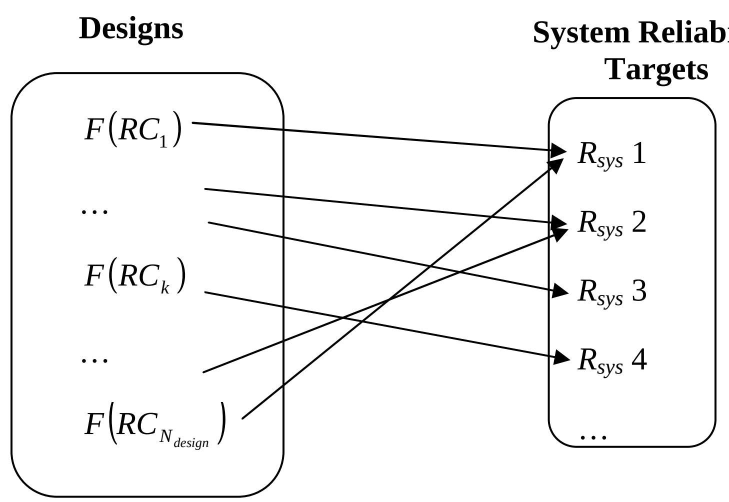
The main contributions of this research are: the development of a methodology for multi-objective optimization to enable reliability-cost tradeoffs and to decide target component reliability for a series system. The method can be extended to non-series system using fault tree analysis for example, to define system failure.

[1] Mileage can be a better measure of "time" for systems that are not constantly in operation, such as a vehicle.

2. MODEL DESCRIPTION

2.1. Component Level Reliability Assignment

If the failure modes of a system are known, component level reliability maps directly to the system level reliability as shown in Figure 1. The number of ways component level reliability can be chosen (designs) for a system composed of a large number of components is very large. As shown in the figure, for N_c components, each having m_{ic} variants (subscript i_c stands for component number), the set of all designs is a Cartesian product of R_{All}^{ic}'s where $R_{All}^{ic} \in \{R_1^{ic}, R_2^{ic}, ..., R_{m_{ic}}^{ic}\}$ and $R_{m_{ic}}^{ic}$ represents the reliability of the m_{ic} variant of the ic^{th} component. Each of these designs can map to a system level reliability using the failure mode information embedded in the functional F, and the mapping is a surjection, i.e. for each feasible system level reliability target there is at least one design. However, more than one design can result in a given system reliability level. Therefore, the inverse problem of "given a system reliability target, allocate component level reliabilities" does not have a unique solution. As we will see later in the paper, the addition of the objective of cost helps us alleviate this issue.



$$RC_{ic} \in R_{All}^{1} \times R_{All}^{2} \times ... \times R_{All}^{N_c}$$
$$R_{All}^{ic} \in \{R_1^{ic}, R_2^{ic}, ..., R_{m_{ic}}^{ic}\}$$
$$N_{design} = \prod_{ic=1}^{N_c} m_{ic}$$

Figure 1. Mapping of component level reliabilities into system level reliability

2.2 System Description

Assume that there is a system of N_c components, which are numbered using $i_C = 1, 2, \ldots, N_c$. We also assume that this system is a series repairable system so that any failure in a component results in a system failure. The failed component can be replaced using a brand new component of the same make, as shown in Figure 2.

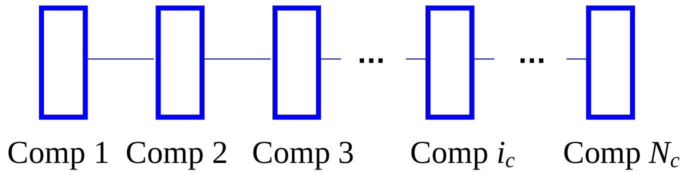


Figure 2. A series system layout

We further assume that the system begins service under similar operational conditions with all brand new components and the mileage M_0 set to zero. For a given system in service, when a component fails, we record the component ID i_C and the odometer mileage reading $\left(M_j\right)_{i_C}$ for that component. The mileage between failures (TBF) is then equal to

$$\mathrm{d}\left(M_j\right)_{i_C} = \left(M_j\right)_{i_C} - \left(M_{j-1}\right)_{i_C}. \tag{1}$$

The failed component is then replaced with a brand new one and the system resumes its regular service. This procedure continues until the odometer mileage reading for the specific system reaches a given threshold value $M_{\mathrm{threshold}}$. Data acquisition for this system is then stopped. Figure 3 shows our notation with the component ID dropped for simplicity.

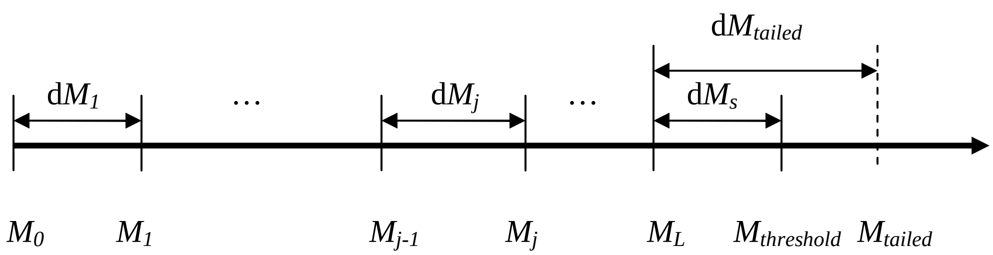


Figure 3. A sorted component failure odometer mileage readings

In Equation (1) and Figure 3, j = 1, 2, …, L is the failure sequence number for the given component. L is the total number of component failures before and at the threshold value $M_{\mathrm{threshold}}$ for the specified component in a system. $M_0 = 0$ is the beginning odometer mileage reading and M_L is the maximum component failure mileage odometer or timer reading, before the censoring mileage, $M_{\mathrm{threshold}}$. Obviously, the recorded component failure data are right-censored at the threshold mileage.

The $(L+1)^{st}$ component to replace the malfunctioned L^{th} component does not fail at or before $M_{\mathrm{threshold}}$ (*surviving* component) but would fail at the odometer mileage M_{tailed} beyond the threshold value $M_{\mathrm{threshold}}$ if the data were not censored. The $(L+1)^{st}$ failure mileage of component i_C given by

$$\mathrm{d}(M_{\mathrm{tailed}})_{i_C} = \mathrm{d}(M_{L+1})_{i_C} = (M_{\mathrm{tailed}})_{i_C} - (M_L)_{i_C} \tag{2}$$

is *not* known from the recorded component failure data. Instead, the surviving component mileage can be retrieved as

$$\mathrm{d}(M_{\mathrm{S}})_{i_{\mathrm{C}}} = M_{\mathrm{threshold}} - (M_L)_{i_{\mathrm{C}}} \tag{3}$$

representing the used mileage/time of the $(L+1)^{st}$ component at the threshold mileage/time $M_{\mathrm{threshold}}$.

Notice that there are cases that

$$\mathrm{d}(M_{\mathrm{S}})_{i_{\mathrm{C}}} = M_{\mathrm{threshold}} \tag{4}$$

indicating that in the component data acquisition process, some components have no failure records at all. This missing information should be included in the surviving component data to make up a total of $N_{\mathrm{Survi}} = N_{\mathrm{C}}$ surviving components for a system. The first failure mileage of the system is defined and obtained as

$$M_{1\mathrm{f}} = \min(M_1)_{i_{\mathrm{C}}} = \min(\mathrm{d}M_1)_{i_{\mathrm{C}}} \tag{5}$$

for $i_{\mathrm{C}} = 1, 2, \ldots, N_{\mathrm{C}}$.

This data acquisition procedure may be applied to a fleet of N_{S} systems of the same make and model, or the simulation may be repeated for the same system N_{S} times. Then, the corresponding records $(M_j)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}}$, $(\mathrm{d}M_j)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}}$ for component i_{c}, are obtained and

$$(M_{1\mathrm{f}})^{i_{\mathrm{S}}} = \min(M_1)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}} = \min(\mathrm{d}M_1)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}} \tag{6}$$

for $i_{\mathrm{S}} = 1, 2, \ldots, N_{\mathrm{S}}$, $i_{\mathrm{c}} = 1, 2, \ldots, N_{\mathrm{c}}$, and $i = 1, 2, \ldots, (L)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}}$. It should be noted that a complete set of system failure data samples can be obtained from actual data or simulations.

2.3 Data Sorting and Parameter Estimation

The originally recorded component failure data are raw data that are usually in a random order and cannot be used directly. Therefore, they need to be sorted according to the priority order of the system or simulation number $i_{\mathrm{S}} = 1, 2, \ldots, N_{\mathrm{S}} = N_{\mathrm{sim}}$, followed by the component serial number $i_{\mathrm{C}} = 1, 2, \ldots, N_{\mathrm{C}}$, and then by the failure sequence $j = 1, 2, \ldots, (L)_{i_{\mathrm{C}}}^{i_{\mathrm{S}}}$.

For a specified component of a specified system or simulation, the sorted component failure odometer mileage readings in ascending sequence, is given by

$$M_0 = 0 < M_1 < \cdots < M_{j-1} < M_j < \cdots < M_L < M_{\mathrm{threshold}} \; . \tag{7}$$

Note that the component failure odometer mileage reading M_j is related to the component mileage between failures (TBF) as

$$M_j = \sum_{k=1}^{j} \mathrm{d}M_k \quad \text{for } j = 1,2,\ldots,L,L+1\,. \tag{8}$$

.

We assume that the mileage between failures (TBF) of all components is a Beta distributed random variable of the same parameters [9-11]. The Beta distribution is used because of its flexibility to model a wide variety of probability distributions. In this case,

$$\mathrm{d}M_i = X_i \sim \beta(A, B, p, q), \quad (A \le X_i \le B, \text{ and } p > 0, q > 0) \tag{9}$$

The probability density function (PDF) of the Beta distribution is given by,

$$\begin{gathered} f(x, A, B, p, q) = \beta(p,q)^{-1} (x - A)^{p-1} (B - x)^{q-1} \big/ (B - A)^{p+q-1} \\ (A \le x \le B, \text{ and } p > 0, q > 0) \end{gathered} \tag{10}$$

and its cumulative distribution function (CDF) is

$$F(x) = P(X_i \le x) = \int_{-\infty}^{x} f(t, A, B, p, q)\mathrm{d}t\,. \tag{11}$$

In Equation (10), the Beta function $\beta(p,q)$ is associated with the Gamma function Γ as,

$$\beta(p,q) = \Gamma(p)\Gamma(q)/\Gamma(p+q) \tag{12}$$

and appears as a normalization constant to ensure that the probability in Equation (11) integrates to unity when $x \to \infty$. The standard Beta distributed random variable Y_i is the special case of the general Beta distributed random variable X_i when $A = 0$ and $B = 1$; i.e.,

$$Y_i \sim \beta(0,1, p, q), \tag{13}$$

where

$$X_i = A + (B - A)Y_i\,. \tag{14}$$

In Equations (9-11, 14), A and B are the lower and upper bounds of the random variable X_i and p and q are the shape parameters. Figure 4 shows the probability density function (PDF) and the corresponding cumulative distribution function (CDF) of a component failure mileage for parameters A = 0, B = 45,000 miles, p = 3, and q = 5.

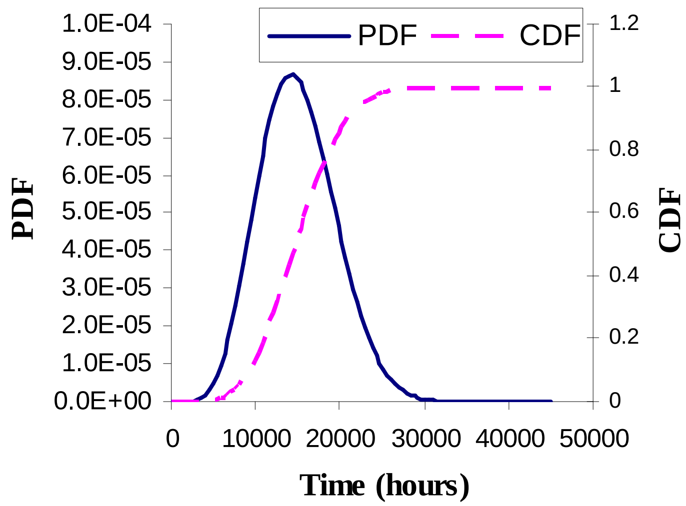


Figure 4. PDF and CDF of a Beta distribution

If we have a total of N_F failure mileage data points x_i, i = 1, 2, ... , N_F the sample mean and the sample standard deviation are $\mu = \frac{1}{N_\mathrm{F}}\sum_{i=1}^{N_\mathrm{F}} x_i$ and $\sigma = \sqrt{\frac{1}{N_\mathrm{F}}\sum_{i=1}^{N_\mathrm{F}}(x_i-\mu)^2}$, respectively. If

$$\bar{\mu} = \frac{\mu - A}{B - A} \tag{15}$$

and

$$\bar{\sigma} = \frac{\sigma}{B - A}, \tag{16}$$

the method-of-moments provides the shape parameters p and q as [10],

$$\begin{aligned} p &= \bar{\mu}\left(\frac{\bar{\mu}(1-\bar{\mu})}{\bar{\sigma}^2} - 1\right), \\ q &= (1-\bar{\mu})\left(\frac{\bar{\mu}(1-\bar{\mu})}{\bar{\sigma}^2} - 1\right) \end{aligned} \quad . \tag{17}$$

If A and B are known or if we have a good initial estimate with uncensored samples, the explicit form method of Equations (15-17) can be used to estimate p and q. It is difficult however, to have a good initial estimate of parameters A and B. Furthermore, the component failure data are usually heavily censored. In this case, the maximum likelihood estimation (MLE) method, which includes the N_Survi surviving mileage samples s_j, j = 1, 2, ..., N_Survi for the component surviving mileages, can be used to estimate A, B, p, and q [12] by maximizing the following likelihood function,

$$L = \prod_{i=1}^{N_{\mathrm{F}}} f\left(x_i, A, B, p, q\right) \prod_{i=1}^{N_{\mathrm{Survi}}} \left[1 - F\left(s_i, A, B, p, q\right)\right]. \tag{18}$$

Conversely, if the Beta parameters *A*, *B*, *p*, and *q* are available for a component, the mean value μ, and standard deviation σ can be calculated as,

$$\mu = A + (B - A)\frac{p}{p+q} \tag{19}$$

and

$$\sigma = \frac{(B-A)}{p+q}\sqrt{\frac{pq}{p+q+1}}\,. \tag{20}$$

A coefficient of variation *COV* can be then defined as

$$COV = \frac{\sigma}{\mu}. \tag{21}$$

The mean value μ, the coefficient of variation *COV*, and the maximum possible failure mileage *B* are more physically meaningful than the Beta parameters *A*, *B*, *p*, and *q*, and therefore, are preferred in our discussions in the remainder of the paper.

2.4 System Reliability Simulation

A system reliability simulation evaluates the system reliability with respect to mileage. In a continuous case, the *non-repairable* system failure rate is defined as

$$\lambda(t) = \frac{f(t)}{R(t)} = \frac{f(t)}{1 - F(t)} \tag{22}$$

where $f(t)$ is the probability density function (PDF) of the time to system failure and $F(t)$ is the cumulative distribution function (CDF) of the time to system failures, calculated by

$$F(t) = \int_{-\infty}^{t} f(t)\mathrm{d}t\,. \tag{23}$$

The system reliability $R(t)$ is given by

$$R(t) = 1 - F(t). \tag{24}$$

Let all simulated failure data being distributed over m bins of a constant width Δt representing mileage (Figure 5). The total number of failures over the m bins is

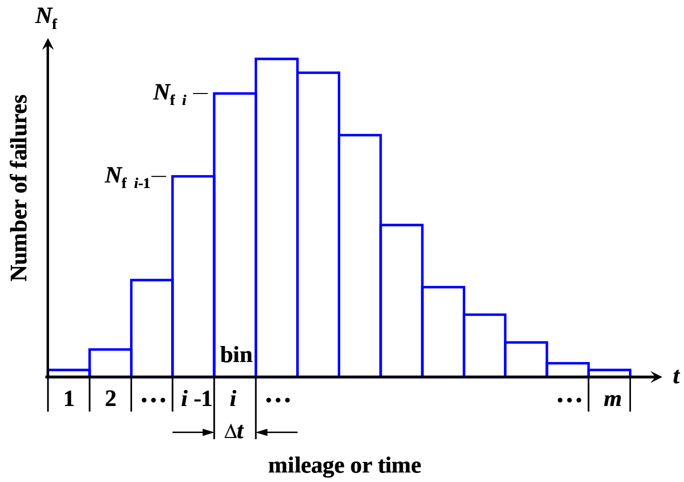


Figure 5. Histogram of failure data

$$N_f = \sum_{i=1}^{m} N_{f_i} \tag{25}$$

and the PDF and CDF values corresponding to bin i are

$$f_i = \frac{N_{f_i}}{N_f \Delta t} \tag{26}$$

and

$$F_i = \sum_{j=1}^{i-1} f_j \Delta t = \sum_{j=1}^{i-1} \frac{N_{f_j}}{N_f \Delta t} \Delta t = \sum_{j=1}^{i-1} \frac{N_{f_j}}{N_f}, \tag{27}$$

respectively. The failure rate for bin i is then provided by

$$\lambda_i = \frac{f_i}{1 - F_i} = \frac{f_i}{1 - \sum_{j=1}^{i-1} \frac{N_{f_j}}{N_f}} = \frac{N_{f_i}}{\left(N_f - \sum_{j=1}^{i-1} N_{f_j} \right) \Delta t} \quad , \tag{28}$$

and the cumulative failure rate is

 UNCLASSIFIED: Dist. A. Approved for public release

$$H_i = \sum_{j=1}^{i} \lambda_j \Delta t \, . \tag{29}$$

The system reliability is then estimated by

$$R_i = 1 - F_i = 1 - \sum_{j=1}^{i-1} \frac{N_{\mathrm{f}_i}}{N_{\mathrm{f}}} \tag{30}$$

or equivalently by

$$R_i = e^{-H_i} \, . \tag{31}$$

It should be noted that Equations (22) to (31) are valid for non-repairable systems (first failure problem). However, they can also provide an approximation of the reliability for repairable systems as well. In this paper, we use quotes for terms as “PDF”, “CDF” or “reliability,” as a reminder that the corresponding quantities are approximate for repairable systems. Our notation for the first failure problem, is changed to $N_{1\mathrm{f}_i}$ for N_{f_i}, $N_{1\mathrm{f}}$ for N_{f}, f_{1i} for f_i, F_{1i} for F_i, λ_{1i} for λ_i, H_{1i} for H_i, and R_{1i} for R_i.

2.5 Reliability-Cost Pareto Front

A Pareto front represents the tradeoffs that must be made in a design-decision problem among multiple objectives (attributes) indicating that improving all attributes simultaneously is not possible. The Pareto front in this study presents tradeoffs between reliability and cost. Its shape gives information on how much additional cost is needed to increase the system reliability by a specified amount. Figure 6 illustrates the shape of the Pareto front for a hypothetical system. The ideal case is to simultaneously increase reliability and reduce cost. The utopia point provides the best such scenario but it is not always attainable. The Pareto front separates the reliability-cost domain into infeasible and feasible sub-domains and provides the most efficient tradeoff strategy.

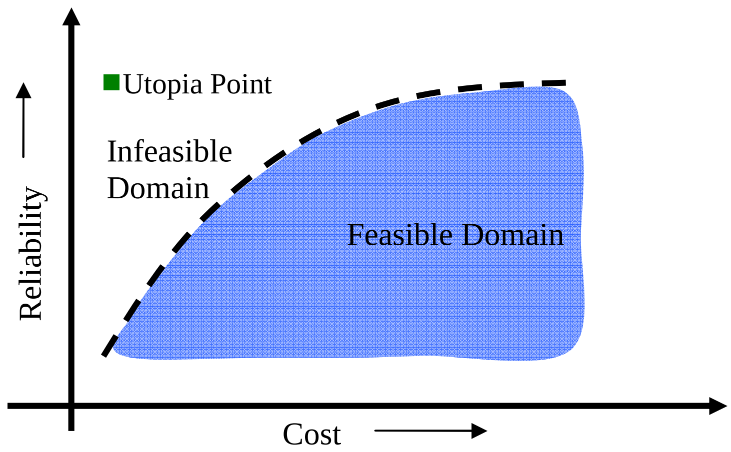


Figure 6. Pareto front between system reliability and associated cost for a hypothetical system

In this paper, a heuristic multi-objective optimization strategy is used to estimate the Pareto front between reliability and the associated cost. To initialize, random points in the design space are generated, evaluated and mapped into the Cost-Reliability space. To efficiently obtain the Pareto front, we evaluate design points only within controlled local domains. Because these local domains are not known *a priori*, they can be determined adaptively if random points are evaluated one by one. The determination of the local domains in the design space is addressed later in the paper. For computational purposes, we discretize the reliability vertex by defining horizontal *slices* in the Cost-Reliability space. We then assume that the evaluated point with the minimum cost within a slice defines a vertex of the approximate Pareto front. The latter is assumed having one vertex per slice.

Figure 7 helps illustrate the local domain and slicing scheme that maps a random design point $\mathbf{x} = (x_1, x_2, \ldots)$ in one of the m - 1 (m = 6 in Figure 7) local domains to a (Cost, R) point in the Cost-Reliability space where m is the number of break points that divide the reliability range [0 1] into m-1 slices (local domains). A diamond marker is used to represent the vertex of the Pareto front and a circular point to represent other designs.

We assume that only the inner m-3 slices, bounded by the reliability values $pareto_L$ and $pareto_U$, are of equal *thickness*. The values of $pareto_L$ and $pareto_U$ are dynamically determined by the current minimum and maximum reliability values, R_{min} and R_{max}, respectively, among all evaluated designs (circular and diamond markers in Figure 7) that are mapped from the design space $\mathbf{x}$ into the Cost-Reliability space.

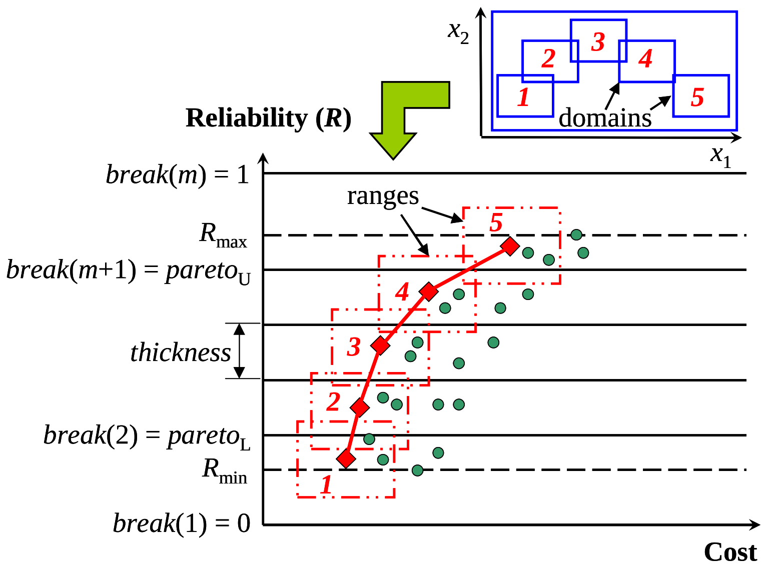


Figure 7. A schematic diagram to illustrate slicing scheme for finding the approximate Pareto front

Assuming that the evaluated points that correspond to R_{min} and R_{max} are always within the bottom and top slices respectively, the relation

$$R_{\min} < pareto_{\mathrm{L}} < pareto_{\mathrm{U}} < R_{\max} \tag{32}$$

holds. We determine the $pareto_L$ and $pareto_U$ values from

$$pareto_{\mathrm{L}} = R_{\min} + \delta_R \text{ and } pareto_{\mathrm{U}} = R_{\max} - \delta_R \tag{33}$$

where δ_R is a *prescribed* small positive real number satisfying

$$0 < \delta_R < \frac{1}{2}\left(R_{\max} - R_{\min}\right). \tag{34}$$

The *thickness* of the i^{th} slice is then provided by

$$thickness(i) = \begin{cases} 1 - pareto_{\mathrm{U}} & \text{for } i = m-1 \\ \dfrac{1}{m-3}\left(pareto_{\mathrm{U}} - pareto_{\mathrm{L}}\right) & \text{for } 2 \le i \le m-2 \\ pareto_{\mathrm{L}} & \text{for } i = 1 \end{cases} \tag{35}$$

and the i^{th} break point is determined by

 UNCLASSIFIED: Dist. A. Approved for public release

$$break(i)=\begin{cases}1 & \text{for} \quad i=m \\ pareto_{\mathrm{L}}+thickness(i-1) & \text{for} \quad 2\le i\le m-1. \\ 0 & \text{for} \quad i=1\end{cases} \tag{36}$$

Equations (32) to (36) determine the Pareto slices based on the current minimum and maximum reliability values, $R_{\min}$ and $R_{\max}$, among all evaluated points. If the reliability of a new point is between the current $R_{\min}$ or $R_{\max}$, the definition of Pareto slices remains the same; otherwise the Pareto slices are re-initialized.

A number of local domains are defined in the design space so that a design point generated in the design space can be mapped onto a point in the Reliability-Cost performance space which is nearby the current Pareto front. In this case, the number of design points to be evaluated will be drastically reduced improving therefore, the computational effort. As seen in Figure 7, a local domain in the design space corresponds to only one slice in the Cost-Reliability space. The center point of the local domain in the design space corresponds to the vertex of the current Pareto front within the corresponding slice. Before any Pareto front vertex is calculated, we assume that each local domain in the design space covers the entire design space (global domain). As a Pareto front vertex is first calculated or updated during the algorithm, the center of the local domain in the design space is updated and the size of the local domain is reduced (e.g. by 15% or 20%). It should be noted that the computational effort to obtain the Pareto front will not be reduced until every slice contains at least one evaluated point. If a slice does not contain an evaluated point (calculated Pareto front vertex), the global domain serves as the corresponding local domain in the design space, and a generated random point in the global domain can be mapped onto any of the Pareto front slices reducing the probability of being mapped onto the slice with no point in it. In this case, we cannot efficiently use randomly selected design points to estimate all segments of the Pareto front.

The algorithm to calculate the approximate Pareto front is described using the following steps.

Step 1. Assign initial values.

i_{point}=0;
$R_{\min}$=1;
$R_{\max}$=0;
$pareto_{\mathrm{L}}$ =0.5;
$pareto_{\mathrm{U}}$ =0.51;
Size of i^{th} local *domain* equals size of global domain for i = 1,…,m-1. The size of a local domain will be reduced to 20% of the global domain size for i = 2,…,m-2, and to 15% of the global domain size for i = 1,m-1, when a point in it is a mapped onto a Pareto front slice.

The above initial values are assigned before any design point is evaluated because the conditions for initializing the Pareto front using Equations (32) to (34) cannot be checked and the local domains in the design space are not known *a priori*.

After the first design point (i_{point} = 1) is evaluated, the values of $R_{\min}$ and $R_{\max}$ are equal. After the second design point (i_{point} = 2) is evaluated, the condition $R_{\min} < R_{\max}$ is satisfied if the reliabilities of the two points are different. After subsequent points are evaluated (i_{point} > 3), the condition $R_{min} < R_{max}$ can be guaranteed. Based on these observations, the robustness of the algorithm is enhanced by using Equations (35) to (36) with $pareto_L = 0.5$ and $pareto_U = 0.51$ if $i_{point} \leq 3$ to define the Pareto slices. Otherwise, the full initialization scheme of Equations (32) to (36) is used.

Step 2. Find a new design point to be evaluated.

Keep generating random points in the design space until a design point **x** falls in one of the *m*-1 local domains. Note that no design point is evaluated until it falls within one of the local domains. No reduction in computational effort is achieved until the size of all local domains is reduced from the original global domain size.

Step 3. Evaluate the new design point.

Set $i_{point} = i_{point}+1$;
Evaluate **x** to obtain the (Cost, R) value.

Step 4. Update R_{min} and R_{max}.

If $R_{min} < R$, then $R_{min} = R$,
If $R_{max} > R$; $R_{max} = R$.

Step 5. Update the Pareto front.

If $i_{point} \leq 3$, then:
    Update the Pareto front using the newly added point i_{point};
Else If $i_{point} > 3$, then:
    If $R_{min} < (pareto_L - thickness)$, then:
        $pareto_L = R_{min} - \delta_R$; ($\delta_R$=0.05)
        Initialize the Pareto slices using Equations 35-36;
        Update the Pareto front using all evaluated points.
    Else If $R_{max} < (pareto_U + thickness)$, then:
        $pareto_U = R_{max} + \delta_R$; ($\delta_R$=0.05)
        Initialize the Pareto slices using Equations 35-36;
        Update the Pareto front using all evaluated points.
    Else
        Update the Pareto front using the newly added point i_{point}.
    End If
End If

Step 6. Stop if the algorithm converges; otherwise, go to Step 2.

The diamond marker in Figure 7 indicates the point with the minimum cost value among all i_{point} points within a slice. Therefore, the complete Pareto front is a polygon that connects the *m*-1 diamond markers. The reliability-cost Pareto set needs the evaluations of both the system reliability and the system cost. The system reliability is evaluated using Equation (30) or Equation (31).

2.6 Definition of Design Variables

In Sections 2.2 and 2.3, we described how to use available data to estimate the probabilistic distribution (PDF) of time (or mileage) between failures (TBF) of each component in a system using the family of Beta distributions. The PDFs of all components are then used to estimate the system reliability using Monte Carlo simulation. The four parameters (A, B, p and q) needed to define a Beta distribution are usually estimated if we know the first four moments of the available data (mean, standard deviation, skewness and kurtosis). Such an estimation may not be accurate however, if we only have limited data. For this reason, we proposed an estimation of the Beta distribution parameters using only the mean and standard deviation (see Equations 15 to 17).

Because it is common in practice to describe component reliability using only the MTBF, we assume in this paper that the coefficient of variation $COV = \sigma / \mu$ is constant, allowing us to calculate the standard deviation if we know the mean. We also assume that the B_{factor} where

$$B_{\text{factor}} = B / \mu = B / \text{MTBF} \tag{37}$$

is constant, resulting in

$$B = B_{\text{factor}} \mu = B_{\text{factor}} \text{MTBF} . \tag{38}$$

In this case, Equations (15) to (17) or Equations (19) to (21) can be used to estimate the four parameters of the Beta distribution using only the MTBF. The latter will then be the only independent design variable for each component.

2.7 Reliability-Cost Relation for Each Component

In order to calculate the Pareto front as described in Section 2.5, we must know how component reliability and cost tradeoff. In this paper, we consider the following exponential relationship between component acquisition cost and its MTBF

$$\text{cost} = \text{cost}_0 \, e^{k(\text{MTBF}/\text{MTBF}_0 - 1)} \tag{39}$$

where k is a cost growth constant which is used to account for different possible relationships between MTTF and cost. It is given by

$$k = \ln(\text{cost}_1 / \text{cost}_0) / (\text{MTBF}/\text{MTBF}_0 - 1) . \tag{40}$$

In Equation (40), cost_0 represents the unit cost to achieve a reliability provided by MTBF_0, and cost_1 represents the unit cost for MTBF_1. It is clear that cost is an increasing function of the MTBF as shown in Figure 8. The relationship is used as an interpolation tool between different component variants, which in practice are discrete values of cost and the associated MTBF.

Although the chosen functional form is a good approximation, it does not limit our methodology. Other cost-MTBF relationships can be easily accommodated.

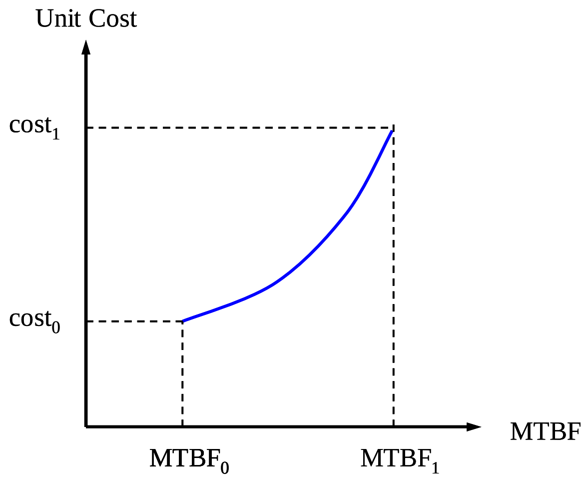


Figure 8. Component cost versus MTBF

Then, for a system of N_C components, the system cost is provided by

$$\text{Cost} = \sum_{i_C=1}^{N_C} \left[\text{cost}_0 \, e^{k(\text{MTBF}/\text{MTBF}_0 - 1)} (1 + \text{failure counts}) \right]_{i_C} . \tag{41}$$

It should be noted that the failure count for each component in Equation (41) can be obtained for both a non-repairable system and a repairable system.

3. EXAMPLE

The approach presented in Section 2 for system cost-reliability simulation and Pareto front generation is demonstrated in this section using a mechanical system consisting of 15 components that are serially connected. The times to failure for all components are statistically independent and Beta distributed. We consider the design of only components 7, 8, 9 and 10, assuming that we actually control their MTBF. Table 1 provides the component data for the system, including the nominal (subscript 0) mean time between failure, the coefficient of variation, the ratio of maximum failure time to the mean time between failures (B_{factor}, in Equation 37), the nominal unit cost (cost of initial design - cost_0, in Equations 39, 40), and the cost growth constant (k, in Equations 39, 40).

Table 1. Component data for the mechanical system

Component Number	*Baseline MTBF in hours ($MTBF_0$)*	*Coefficient of Variation*	B_{factor}	*Baseline cost ($Cost_0$)*	*k*
1	4076	0.3	3	$27,500.00	1
2	15000	0.3	3	$7,000.00	1
3	26510	0.3	3	$3,000.00	1
4	40000	0.3	3	$5,000.00	1
5	18000	0.3	3	$5,000.00	1
6	8000	0.3	3	$500.00	1
7	31809	0.3	3	$22,500.00	1
8	9520	0.3	3	$30,000.00	1
9	9713	0.3	3	$12,500.00	1
10	2330	0.3	3	$20,000.00	1
11	40000	0.3	3	$27,500.00	1
12	8614	0.3	3	$1,000.00	1
13	45000	0.3	3	$30,000.00	1
14	20000	0.3	3	$3,000.00	1
15	25000	0.3	3	$15,000.00	1

A complete set of system failure data samples is obtained using simulation as described in Section 2.2. The system threshold time and truncated threshold time for the data collection are five and one times respectively, the minimum of the maximum failure times of all 15 components.

The time to first failure and all subsequent failure times up to the system threshold time are recorded for 4000 simulations and used for system reliability calculations based on Equations (25) to (31). The specified useful time for reliability calculations is 2500 hours. The component and system cost are calculated using Equations (39) to (41). The simulation results are presented below.

Figure 9 shows the repairable system failure frequency distribution over 200 bins of a constant bin width of Δt = 35 hours over the system truncated threshold time of 6990 hours.

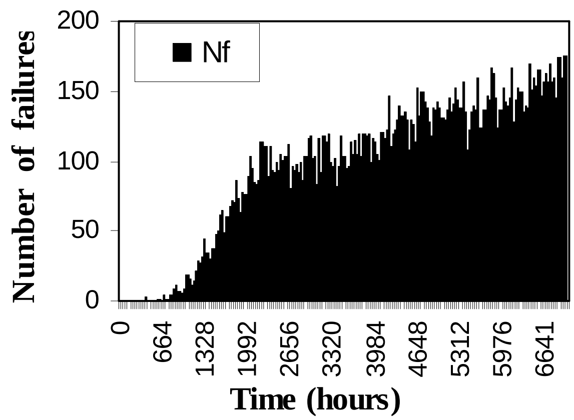


Figure 9. Histogram of **s**ystem failures

Figure 10 shows the "PDF" using Equation (26) for both the repairable system failure data (f in the figure) and the system first failure data (f1 in the figure), as well as the Beta distribution fitting of the repairable system failure data using the maximum likelihood approach of Equation (18). It should be noted that the "PDF" for a repairable system as calculated using Equation (26), is not a real PDF because the failure components are replaced with brand new ones after failure. It simply provides a measure of the frequency of failures for a repairable system.

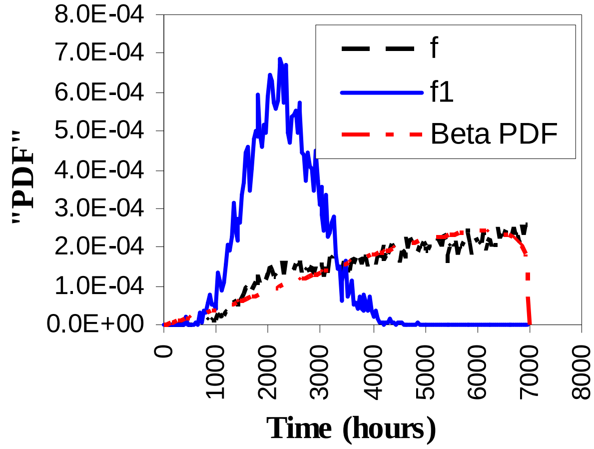


Figure 10. "PDF" comparison

Figure 11 compares the repairable system failure rate λ with the failure rate $\lambda 1$ using only first failure data. As expected, the failure rate of the non-repairable (first failure) system is higher than the failure rate of the repairable system.

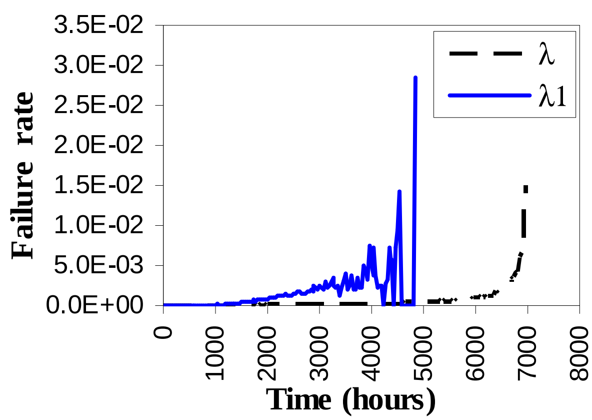


Figure 11. System failure rate comparison

 UNCLASSIFIED: Dist. A. Approved for public release

Figure 12 shows the cumulative distribution function ("CDF") and the cumulative failure rates (H) using repairable system failure data (curves F and H) and the first failure system data (curves F1 and H1). The "CDF" curves F and F1 are calculated using Equation (27) and the cumulative failure rate curves H and H1 are calculated using Equation (29). As expected, the results are drastically different between the repairable and non-repairable systems.

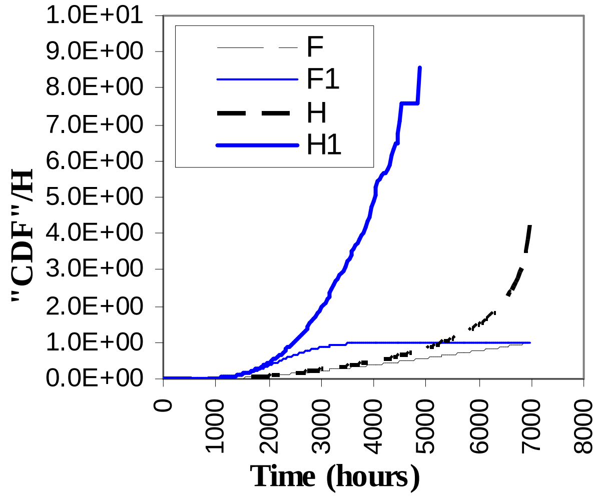


Figure 12. "CDF"/H comparison

Figure 13 compares the reliabilities calculated by Equations (30) and (31). As expected, both equations provide the same reliability results. The RF (Equation 30) and RH (Equation 31) curves (repairable system) and the RF1 and RH1 curves (non-repairable system) are identical. However, the reliabilities are different between the repairable and non-repairable systems.

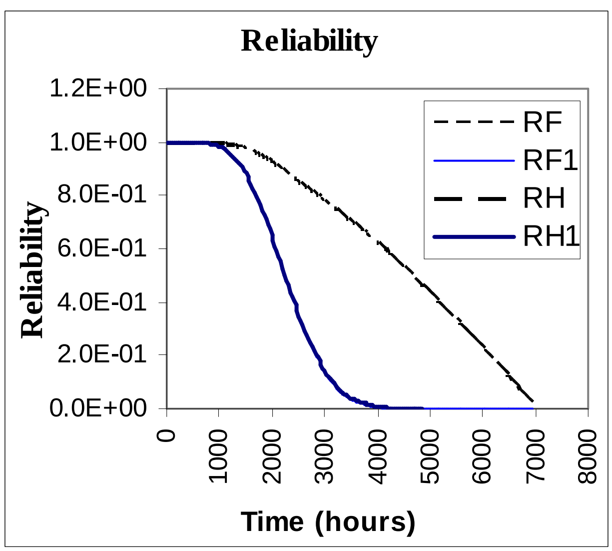


Figure 13. System reliability comparison for repairable and non-repairable systems

Figure 14 shows the first failure count per system simulation for each component and Figure 15 shows the corresponding cost for each component from Equation (39). Although most of failures per system simulation are recorded for components 10 and 1 (Figure 14), the expected cost from all failures is high for components 1, 7, 8, 10, 11 and 13 (Figure 15). This is due to the relatively high component cost for components 1, 7, 8, 10, 11 and 13 (see Table 1).

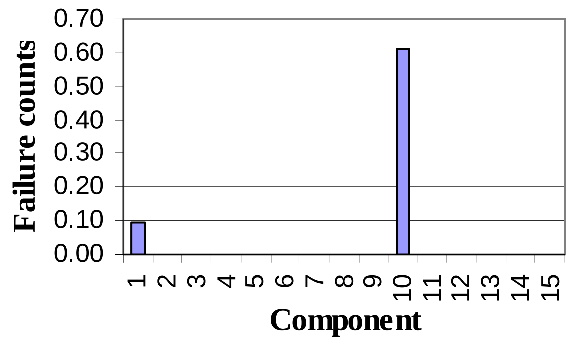


Figure 14. First failure counts for each component per system simulation

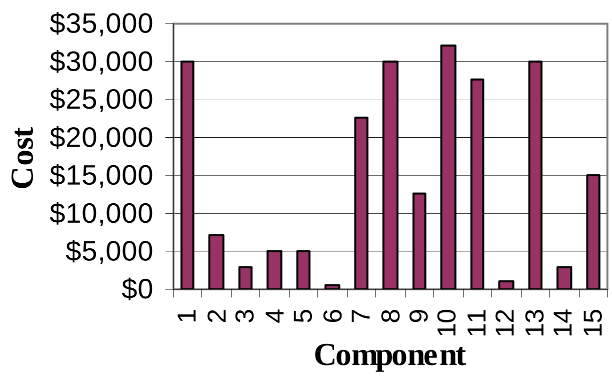


Figure 15. Total cost for each component per system simulation

Finally, Figure 16 shows the cost versus system reliability Pareto front using first failure data. Among the 15 components, only components 7, 8, 9 and 10 are designed. A total of 250 system design points were used to generate the Pareto front considering only 7 slices. The multi-objective optimization strategy presented in Section 2.5 asymptotically approaches the Pareto front between reliability and associated cost. The Pareto front can be presented to the end user who can then select the best point on the front based on his/her tradeoff preferences. For each point on the Pareto front, we have an associated design in terms of target component level reliabilities and costs which would result in the given system reliability and cost.

It is observed from Figure 16 that there is a high slope region where the reliability can be significantly increased with a small increase in cost. This indicates that we can increase the system reliability with a very small increase in cost.

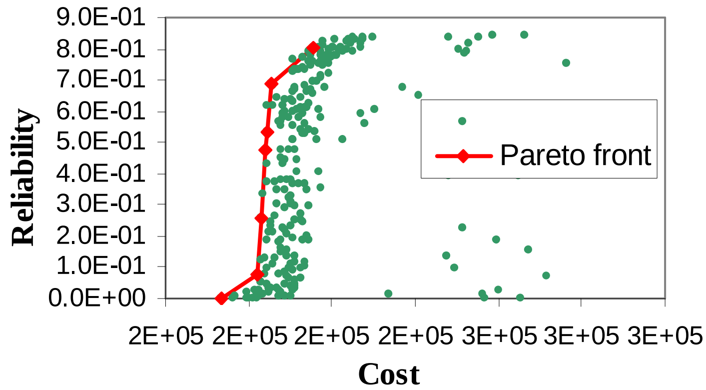


Figure 16. System reliability-cost Pareto front

4. SUMMARY AND CONCLUSIONS

Reliability analysis is critical to understanding the design and maintainability of complex systems. In this paper a simulation based reliability-cost modeling capability was formulated and presented. The methodology accomplishes two things:

1) System Reliability-Cost Simulation: The method uses the Probability Density Function (PDF) of the Time Between Failures (TBF) for each component of the system. The user supplies a MTBF and a Coefficient of Variation (*COV*) for each component, and the presented approach calculates the component PDFs/CDFs of the times between failure. Subsequently, a Monte Carlo simulation estimates the system failure rate, $\lambda(t)$, and the reliability, $R(t)$, as a function of time. The number of failures for each component is also calculated and the associated cost to replace failed components. The latter is used to calculate the required cost for a desired reliability level.

2) Generation of Reliability-Cost Tradeoff: A reliability-cost tradeoff analysis is performed using a user-supplied relationship between reliability and cost for each component. The reliability-cost tradeoff is used to optimally determine the reliability of each component in order to maximize the system reliability and simultaneously minimize the acquisition and repair cost for the system. A heuristic multi-objective optimization algorithm has been developed to calculate the Pareto front between reliability and associated cost.

An example of a mechanical system of 15 serially connected components was used to illustrate the methodology. The example clearly highlighted the differences between repairable and non-repairable systems in terms of PDFs of time between failures, and failure rates. It also demonstrated how the proposed method can be used to efficiently generate the Pareto front for Reliability-Cost tradeoff, which the end user selects the best design from. Each point on the Pareto front is mapped to a target component-level reliability and cost which result in the given system level reliability and cost.

The presented work can be easily extended to a general (not serial) system where a fault tree for example, can define the system failure. In this work, we assumed that the reliability-cost relationship is explicitly known. Future work can attempt to further understand this relationship and accommodate the effect of uncertainty. Finally, the Pareto front analysis can be carried out using other than the reliability and cost attributes.

ACKNOWLEDGMENT

This material is based upon work supported by the U.S. Army TACOM Life Cycle Command under Contract No. W56HZV-08-C-0236, through a subcontract with Mississippi State University, and was performed for the Simulation Based Reliability and Safety (SimBRS) research program. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Army TACOM Life Cycle Command.

REFERENCES

1. Li, J. and Mourelatos, Z. P., “Time-Dependent Reliability Estimation for Dynamic Problems using a Niching Genetic Algorithm,” *ASME Journal of Mechanical Design*, 131(7), 2009.
2. Rackwitz, R., and Fiessler, B., “Structural Reliability under Combined Random Load Sequences,” *Computer and Structures*, 9(5), 484-494, 1978.
3. Der Kiureghian, A., Lin, L. Z., and Hwang, S. J., “Second-Order Reliability Approximation,” *ASCE Journal of Engineering Mechanics*, 113(8), 1208-25, 1987.
4. Zhao, Y. and Ono, T., “A General Procedure for First/Second-Order Reliability Method (FORM/SORM),” *Structural Safety*, 21(2), 95-112, 1999.
5. Der Kiureghian, A. and Dakessian, T., “Multiple Design Points in First and Second-Order Reliability,” *Structural Safety*, 20(1), 37-49, 1998.
6. Du, X., and Chen, W., “Sequential Optimization and Reliability Assessment Method for Efficient Probabilistic Design,” *ASME Journal of Mechanical Design*, 126(2), 225-233, 2004.
7. Liang, J., Mourelatos, Z. P., and Tu, J., “A Single-Loop Method for Reliability-Based Design Optimization,” *International Journal of Product Development*, 5(1/2), 76-92, 2008.
8. Goldberg, D. E., “Genetic Algorithms in Search, Optimization and Machine Learning” Addison-Wesley Publishing Company, 197-199, 1989.
9. Meeker, W.Q. and Escobar, L.A., *Statistical Methods for Reliability Data*, John Wiley & Sons, 1998.
10. Lehmann, E. L. and Casella, G., *Theory of Point Estimation*, Springer, 2nd ed., 1998.
11. Rausand, M. and Hoyland, A., *System Reliability Theory: Models, Statistical Methods, and Applications, 2nd Edition*, Wiley Series in Probability and Statistics, John Wiley & Sons, Inc., Hoboken, New Jersey, 2004.
12. Aldrich, J., "R.A. Fisher and the Making of Maximum Likelihood 1912-1922", *Statistical Science* 12 (3), 162-176, 1997.

CONTACT INFORMATION

Zissimos P. Mourelatos, Ph.D.
Mechanical Engineering Department
Oakland University

Rochester, MI 48309
Tel: 248-370-2686
mourelat@oakland.edu

Jing Li, Ph.D.
Mechanical Engineering Department
Oakland University
Rochester, MI 48309
li2@oakland.edu

Vijitashwa Pandey, Ph.D.
Mechanical Engineering Department
Oakland University
Rochester, MI 48309
pandey2@oakland.edu